

الگوریتم کلی

الگوریتم پایه ان بر اساس مسئله box stacking است

توضیح کد

در کلاس box به ازای هر مکعبی که ساخته میشه ویژگی های از جمله ارتفاع طول و عرض و وزن مکعب رنگ 6 وجه مکعب (جلو عقب چپ راست بالا پایین) برای هر ابعجت ساخته میشه

متد `print Max Stack Height`:

در این متد ابتدا دو ورودی می گیره که اولاً ارایه ای از مکعب هارو می گیره سپس تعداد مکعبی که کاربر وارد کرده. اما کاری که ما می کنیم تعداد کل جعبه هایی که در نظر میگیریم رو ضرب در 6 می کنیم به این دلیل اگر فرض کنیم تعداد مکعب های ما 4 تا باشد یک وجه رو به رو ما هست یک وجه سمت چپ یک وجه سمت راست و یک وجه پشت ما قرار دارد پس ما در واقع 24 حالت مختلف داریم حال اگر ما 180 درجه بچرخونیم از اونجایی که رنگ پایین مکعب و طول و عرض وجه پایین برامون مهم پس در واقع تعداد حالات کاهش پیدا می کند به 12 تا. حالا اگر 90 درجه بچرخونیم حالتی که پیش روی ما است یک حالت و با چرخش 90 درجه دیگه یک حالت دیگه بدست میاد اما نکته اینجا است که توی این حالت با اینگه رنگ وجه پایین با بالا برابر است اما طول و عرض تغییر می کند منظور مساحت کف است. ما این حالت رو هم حذف می کنیم (در چک کردن طول و عرض مکعب بالا و پایین به این صورت عمل می کنیم که طول پایین با عرض بالا و عرض پایین با طول بالا چک می کنیم) و در نهایت 6 حالت برای هر وجه داریم.

ارایه از box ها می سازیم به اسم `allBoxrotation` که همه حالات ممکن برای چرخش مکعب هارو ذخیره می کند

ارایه `boxMaxHeightorder` برای هر جعبه ای که می گیم مسیر رسیدن برای کسب بلند ترین ارتفاع بهمون میگه در واقع این ارایه چرخش های مورد نیاز برای کسب بلند ترین ارتفاع رو بهمون میده

در `for` اول تمام حالت هایی که بالا توضیح داده شد و چگونگی حذف حالات رو در این قسمت انجام می دیم سپس ارایه رو برا اساس وزن جعبه ها سورت می کنیم از بزرگ به کوچک و هم چنین ایندکس ارایه هارو با توجه به چرخشی که انجام می دیم عوض می کنیم

در for سوم که تو در تو است در واقع dynamic programming استفاده شد به این صورت که از سنگین ترین مکعب به سمت کوچک ترین مکعب حرکت می کنیم سپس چک می کنیم جعبه ای که می خواهیم رو جعبه دیگری قرار بدهیم با توجه به شرایط رنگ دو مکعب و وزن آنها می توانیم قرار بدهیم یا نه؟

پیچیدگی

پیچیدگی حافظه $O(n)$

پیچیدگی زمانی $O(n^2) = n(n-1)/2$

خروجی

```
BoxStacking
"C:\Program Files\Java\jdk1.8.0_281\bin\java.exe" ...
Enter cubes count: 3

Box #1:
Enter height: 10
Enter width: 1
Enter depth: 2
Enter weight: 3
Enter front color in RGB format (255,255,255): 1,2,3
Enter back color in RGB format (255,255,255): 1,2,3
Enter left color in RGB format (255,255,255): 1,2,3
Enter right color in RGB format (255,255,255): 1,2,3
Enter up color in RGB format (255,255,255): 1,2,3
Enter down color in RGB format (255,255,255): 0,0,0

Box #2:
Enter height: 26
Enter width: 8
Enter depth: 9
Enter weight: 10
Enter front color in RGB format (255,255,255): 1,2,3
Enter back color in RGB format (255,255,255): 1,2,3
Enter left color in RGB format (255,255,255): 1,2,3
Enter right color in RGB format (255,255,255): 1,2,3
Enter up color in RGB format (255,255,255): 255,255,255
Enter down color in RGB format (255,255,255): 1,2,3
```

```
Box #3:
Enter height: 15
Enter width: 3
Enter depth: 4
Enter weight: 5
Enter front color in RGB format (255,255,255): 1,2,3
Enter back color in RGB format (255,255,255): 1,2,3
Enter left color in RGB format (255,255,255): 1,2,3
Enter right color in RGB format (255,255,255): 1,2,3
Enter up color in RGB format (255,255,255): 255,255,255
Enter down color in RGB format (255,255,255): 0,0,0

Max stack height: 51
Route: #2#3#1

Process finished with exit code 0
```